

Sistem Inspeksi Kecacatan pada Kaleng Menggunakan Filter Warna HSL dan *Template Matching*

Budi Sugandi*, Sintya Dewi

Jurusan Teknik Elektro

Politeknik Negeri Batam

Batam

*budi_sugandi@polibatam.ac.id

Abstrak-Dalam industri makanan dan minuman dengan kemasan kaleng, pengecekan atau inspeksi kecacatan kaleng merupakan bagian penting dalam menjaga kualitas sebuah produk. Inspeksi tersebut sebagian masih menggunakan operator manual yang bergantung pada penglihatan para pekerja dengan segala keterbatasannya. Untuk menanggulangi masalah tersebut, artikel ini mengusulkan proses inspeksi kecacatan menggunakan filter warna HSL dan *template matching* berbasis *contour analysis* yang dilakukan secara otomatis tanpa bantuan operator. Proses inspeksi diawali dengan pengambilan citra RGB oleh kamera. Citra RGB tersebut kemudian dikonversi ke citra aras keabuan (*gray level*). Untuk dapat mendeteksi kecacatan, penulis memakai filter *luminance* dari ruang warna HSL. Proses *template matching* diterapkan pada masing-masing citra dengan cara membandingkan *contour* citra *template* sebagai *contour* citra referensi dengan *contour* citra uji. Kemasan kaleng yang digunakan sebagai *template* mempunyai diameter 4 cm. Dengan menggunakan kriteria uji jarak masing-masing *pixel* pada tiap lingkaran kaleng dengan lingkaran pada kaleng *template*, kaleng diklasifikasikan menjadi dua kategori yaitu kaleng OK (*Good*) atau *reject*. Hasil pengujian menunjukkan, dengan menerapkan nilai *threshold* sebesar 4 cm, nilai RMS untuk objek kategori OK adalah 0,59. Sementara untuk objek kategori *reject* didapatkan nilai RMS sebesar 19,59; 5,05 dan 15, 05 berturut-turut untuk objek *reject* karena diameter tidak beraturan, objek *reject* dengan pembuka kaleng terbuka dan objek *reject* tanpa pembuka kaleng.

Kata kunci: inspeksi kecacatan; filter warna HSL; *template matching*, *contour analysis*

1. Pendahuluan

Dalam kehidupan sehari-hari, kita sering mengonsumsi bermacam-macam makanan atau minuman dengan kemasan kaleng. Setiap kemasan kaleng dalam satu produk pasti memiliki bentuk yang sama. Pengecekan kecacatan pada kemasan kaleng sebagian besar dilakukan secara manual menggunakan operator manusia yang prosesnya bergantung pada penglihatan mata para pekerja. Penglihatan manusia tentu saja memiliki keterbatasan akibat kelelahan maupun kekurangtelitian dalam pengecekan, sehingga membutuhkan tenaga pekerja yang banyak untuk memastikan akurasinya.

Beberapa studi telah dilakukan untuk mengatasi permasalahan di atas. Salah satunya dengan menggunakan deteksi kelainan tekstur objek yang akan dibandingkan [1]. Dalam studinya, penelitian ini mengklasifikasi objek berdasarkan objek rusak atau tidak. Proses pengklasifikasian menggunakan metode *Principal Component Analysis* (PCA). Sementara itu untuk menjamin kualitas kaleng agar produk yang dikemas terstandarisasi dan untuk meminimisasi kerusakan kaleng, pengontrolan kualitas kaleng menjadi sangat penting. Klasifikasi kaleng jenis *reject* pada beberapa ukuran volume kaleng juga pernah dilakukan sebelumnya [2]. Metode yang digunakan

adalah *Statistical Process Control* (SPC) dan *Statistical Quality Control* (SQC).

Pendeteksian objek berwarna sering kali menggunakan filter warna, baik itu filter warna RGB, HSL/HSV, maupun YCbCr [3]-[11]. Filter warna HSL/HSV banyak digunakan oleh banyak peneliti khususnya di bidang *object recognition* [3], *machine vision* [11] maupun *robot vision* [8], [10]. Penggabungan pendeteksian filter warna HSL dengan partikel filter sebagai metode untuk pelacakan objeknya berhasil memisahkan citra wajah dari citra latar dengan mengatur nilai filter warna HSL[3]. Sementara penggabungan metode pendeteksian objek dengan filter warna HSV dengan metode klasifikasi KNN (*K Nearest Neighbor*) dilakukan untuk mengklasifikasi objek berdasarkan data latih yang dibandingkan dengan data uji yang ada[11]. Penelitian ini berhasil mendeteksi dan mengklasifikasikan objek dengan tingkat akurasi 88.75% untuk objek tanpa citra latar dan 77,35% untuk objek dengan citra latar. Dalam bidang *robot vision*, filter warna HSV digunakan untuk mendeteksi gawang [10]. Hasil dari pendeteksian gawang akan menjadi referensi bagi robot untuk menentukan lokasi dirinya terhadap gawang.

Metode *template matching* merupakan metode untuk membandingkan citra referensi (*template*) dengan citra uji. *Template matching* telah banyak digunakan dalam beberapa

studi di bidang biometrik [12]-[14] maupun bidang pengenalan objek (*object recognition*) [15]-[17]. Metode *template matching* untuk biometrik dengan membandingkan ciri pada citra uji dan citra referensi telah dilakukan oleh [12]. Penelitian ini membandingkan beberapa ciri dalam satu citra wajah uji dengan ciri-ciri citra referensi. Ciri citra yang digunakan adalah nilai intensitas dari *pixel* tiap citra wajah. Metode ini banyak kelemahan karena citra wajah sangat sensitive terhadap cahaya. Perhitungan *cross correlation* antara citra uji dan citra referensi menunjukkan hasil yang menjanjikan [14]. Hanya saja dengan citra uji dan referensi yang banyak akan memerlukan waktu pemrosesan yang lama. Selain itu untuk beberapa kasus mempunyai akurasi yang rendah. Sementara itu untuk penggunaan *template matching* pada pengenalan objek telah dilakukan [17]. Penelitian ini menggabungkan *template matching* dengan algoritma *Artificial Bee Colony*. Penelitian ini berhasil mendeteksi objek dengan akurasi 100% dan berhasil mengoptimasi waktu pemrosesan.

Untuk mengatasi beberapa kekurangan dari metode yang telah digunakan oleh beberapa penelitian di atas, penelitian ini mengusulkan proses inspeksi kecacatan kaleng dengan mengaplikasikan filter warna HSL dan *template matching* berbasis *contour analysis* yang dilakukan secara otomatis tanpa bantuan operator. Filter warna HSL dipilih karena filter HSL merupakan filter warna yang tidak sensitif terhadap perubahan cahaya di sekitar objek. Filter warna HSL akan digunakan untuk mendeteksi kaleng dan memisahkan citra kaleng dari citra latar. Sementara *template matching* berbasis *contour analysis* digunakan untuk mengklasifikasi kaleng menjadi kaleng dengan kategori OK atau *reject*. Selain karena metode *template matching* ini mempunyai waktu pemrosesan yang relatif cepat, juga mempunyai tingkat akurasi yang tinggi. Dengan menggabungkan kedua metode tersebut, tingkat keakuratan pengujian akan menjadi lebih baik.

Selanjutnya untuk memudahkan pembahasan, artikel ini disusun sebagai berikut. Sesi dua berisikan metode penelitian yang penulis lakukan. Hasil pengujian dan pembahasan ditempatkan di sesi tiga. Terakhir kesimpulan dipaparkan di sesi empat.

1. Metode

Proses inspeksi diawali dengan pengambilan citra oleh kamera. Citra yang tertangkap kamera merupakan citra RGB yang akan dikonversi ke citra aras keabuan (*gray level*). Untuk dapat mendeteksi kecacatan kaleng, penulis memakai filter *luminance* dari ruang warna HSL. Proses *template matching* diterapkan pada masing-masing citra dengan cara membandingkan citra *template* sebagai citra referensi dengan citra uji. *Template* yang digunakan adalah bentuk lingkaran dari kaleng yang tampak dari atas. Data *template* yang disimpan berupa data titik pada setiap keliling lingkaran. Dengan membandingkan jarak antara diameter kaleng *template* pada tiap titik uji dengan kaleng sampel, kaleng diklasifikasikan menjadi dua kategori yaitu kaleng OK (*Good*) atau kaleng *reject*. Kriteria uji yang digunakan adalah jarak masing-masing *pixel* pada tiap lingkaran kaleng dengan lingkaran pada kaleng *template*.

a. Citra RGB (Red, Green dan Blue)

Sebuah citra didefinisikan sebagai fungsi dua dimensi

dari pasangan koordinat yang menyatakan posisi dari titik citra dan nilai intensitas atau *gray level* dari citra di titik tersebut. Jika nilai intensitas atau derajat keabuan citra tersebut dilakukan proses kuantisasi dan pada pasangan koordinat dilakukan proses sampling, maka hasil dari kedua proses tersebut akan menghasilkan citra digital. Citra digital terdiri dari elemen-elemen gambar atau *pixel* (*picture elemen*) yang mempunyai lokasi dan intensitas tertentu. Citra RGB adalah citra asli hasil penangkapan kamera. Citra RGB mempunyai tiga kanal warna *Red*, *Green* dan *Blue* yang masing-masing mempunyai 256 level warna dari 0–255.

b. Citra Aras Keabuan (Gray Scale Image)

Citra keabuan merupakan citra yang nilai intensitas pikselnya didasarkan pada derajat keabuan. Pada citra keabuan, derajat warna hitam sampai dengan putih dibagi ke dalam 256 derajat keabuan di mana warna hitam sempurna direpresentasikan dengan nilai 0 dan putih sempurna dengan nilai 255.

Citra RGB dapat dikonversi menjadi citra keabuan yang hanya mempunyai satu kanal warna. Persamaan yang umumnya digunakan untuk mengkonversi citra RGB menjadi citra keabuan ditunjukkan pada rumus (1).

$$\text{citra keabuan} = 0.298 * R + 0.587 * G + 0.114 * B \quad (1)$$

c. Filter Citra HSL

Warna HSL (*Hue, Saturation, Luminance*) merupakan salah satu warna yang digunakan untuk memfilter (membedakan) objek yang satu dari objek lainnya. Warna HSL banyak digunakan sebagai filter objek berwarna terutama jika keadaan latar yang berubah-ubah karena pengaruh cahaya. Dibanding warna aslinya RGB, warna HSL lebih mudah digunakan untuk membedakan objek yang satu dengan objek lainnya. Gambar 1 menunjukkan contoh konversi citra dari RGB ke HSL.

Konversi warna RGB ke warna HSL ditunjukkan dalam rumus (2).

$$r = \frac{R}{255}; g = \frac{G}{255}; b = \frac{B}{255}$$

$$d = \max(r, g, b) - \min(r, g, b)$$

$$L = \frac{\max(r, g, b) + \min(r, g, b)}{2} \quad (2)$$

$$H = \begin{cases} 0, & \text{if } d = 0 \\ \frac{60(G - B)}{d}, & \text{if } \max = r \\ 120 + \frac{60(B - R)}{d}, & \text{if } \max = g \\ 240 + \frac{60(R - G)}{d}, & \text{if } \max = b \end{cases}$$

$$S = \begin{cases} 0, & \text{if } d = 0 \\ \frac{d}{\max(r, g, b) + \min(r, g, b)}, & \text{if } L < 0.5 \\ \frac{d}{2 - \max(r, g, b) - \min(r, g, b)}, & \text{if } L > 0.5 \end{cases}$$

Besarnya nilai dari HSL ini adalah sebagai berikut: $0 \leq L \leq 1$, $0 \leq S \leq 1$, dan $0 \leq H \leq 360^\circ$.



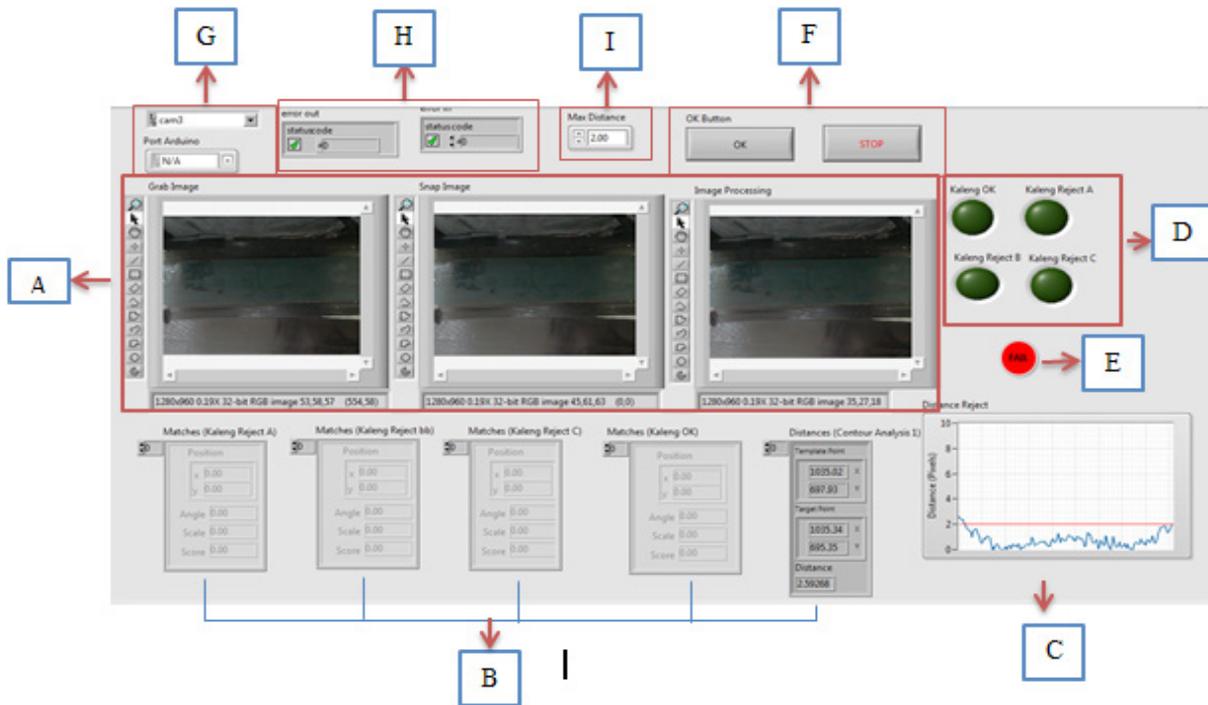
a. Original image

b. HSL image

Gambar 1. Konversi Citra RGB ke HSL



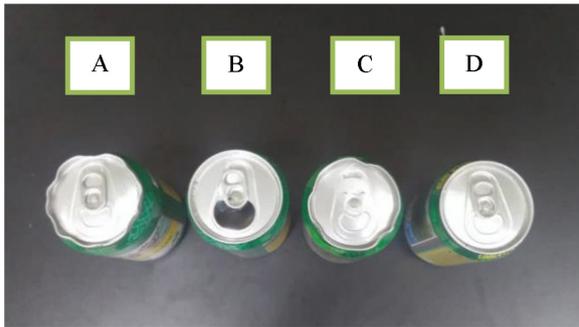
Gambar 2. Ekstraksi template

Gambar 3. Tampilan *interface* aplikasi

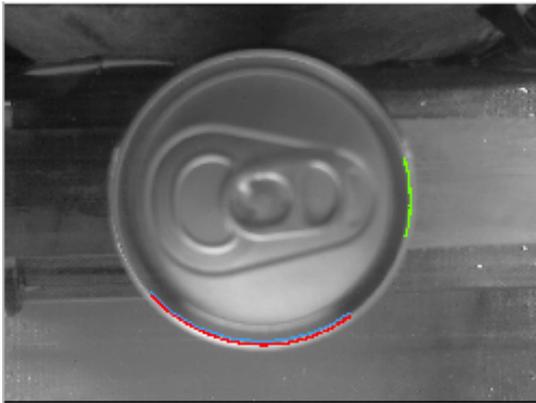
d. Template Matching

Template matching adalah suatu proses membandingkan suatu citra *template* (referensi) dengan citra uji dengan suatu kriteria *matching*. Bagian ini merupakan bagian kunci yang menentukan keberhasilan proses pengklasifikasian kaleng. Dalam artikel ini digunakan *template matching* berbasis *contour analysis*, yaitu *template matching* yang didasarkan pada penghitungan kemiripan *contour* antara objek yang diuji dengan objek referensi. Proses *template matching* dilakukan setelah didapatkan citra objek yang akan diuji. Citra objek ini didapatkan dari proses filter warna HSL. Proses *template*

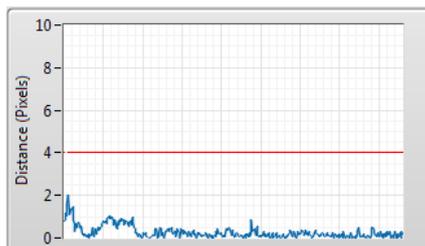
matching didahului dengan proses ekstraksi *template* untuk mendapatkan data ciri dari *template* yang dibutuhkan sebagai ciri referensi. Gambar 2 menunjukkan hasil ekstraksi titik-titik pada keliling lingkaran kaleng sebagai ciri *contour* yang akan dianalisis kemiripannya. Garis merah menunjukkan kumpulan titik-titik uji pada *template* di keliling lingkaran. Setelah *template* diekstraksi, ciri tersebut disimpan dalam *database* yang akan dijadikan sebagai *template*. Ciri objek disimpan dalam bentuk jarak titik-titik dalam garis merah terhadap pusat lingkaran dalam kaleng sebagai *contour* yang akan dianalisis.



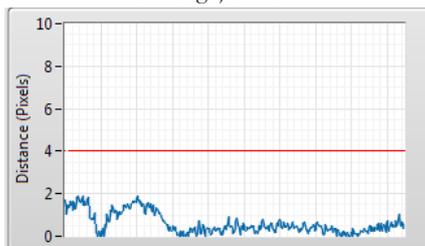
Gambar 4. Sample uji.



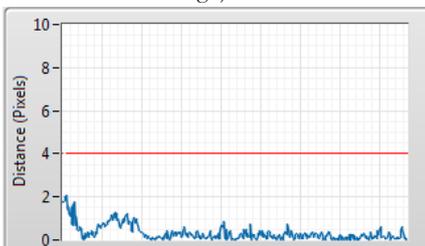
Gambar 5. Sampel uji pada kondisi GOOD (OK)



a. Pengujian ke 1



b. Pengujian ke 2



c. Pengujian ke 3

Gambar 6. Hasil pengujian objek OK

Langkah selanjutnya adalah menghitung *matching score* pada tiap titik uji antara kaleng uji dan kaleng *template*. *Matching score* dihitung berdasarkan nilai RMS (*Root Mean*

Square) dari jarak tiap titik pada kaleng uji ke titik pada *template* dihitung berdasarkan rumus (3).

$$x_{rms} = \sqrt{\frac{\sum_{i=1}^N x_i^2}{N}} \quad (3)$$

dengan x adalah jarak masing-masing titik ke titik uji *template*. Objek yang *matching* akan mempunyai nilai RMS di bawah *threshold* yang telah ditentukan

3. Hasil

Gambar 3 menunjukkan tampilan *interface* aplikasi yang dibuat berdasarkan *NI Labview Vision* yang terdiri dari *A* adalah 3 buah *picture box* yang berfungsi untuk mengakuisisi *frame* video, menampilkan gambar dan menampilkan hasil pengolahan gambar, *B* adalah tampilan untuk memonitoring data yang keluar pada *template matching*, *C* adalah grafik yang keluar pada *template matching*, *D* adalah indikator untuk membedakan jenis kaleng yang OK atau *reject*, *E* adalah lampu status kaleng OK atau *reject*, *F* adalah tombol OK untuk meng-*capture* gambar, *G* adalah *combo box* untuk memilih *port* kamera dan *arduino* yang digunakan, *H* adalah indikator *error* apabila sistem mengalami kesalahan, *I* adalah nilai maksimum *distance pixels* dalam menentukan nilai kaleng OK atau *reject*.

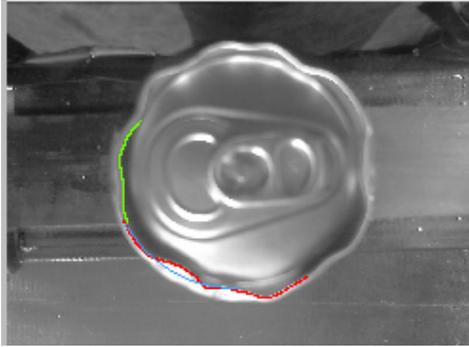
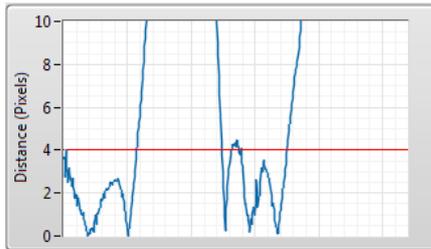
Sampel uji ditunjukkan oleh gambar 4. *A* adalah kaleng *reject1* dengan diameter yang tidak sesuai. *B* adalah kaleng *reject2* dengan pembuka kaleng terbuka. *C* adalah kaleng *reject3* dengan diameter yang tidak beraturan dan tidak memiliki pembuka kaleng. *D* adalah kaleng dengan kondisi baik (OK).

Dalam penelitian ini, penulis menggunakan *template matching* berbasis *contour analysis* untuk menentukan kriteria *reject* pada kaleng. *Contour analysis* akan menghitung kemiripan dua objek dengan menghitung jarak masing-masing titik terhadap titik uji. Dalam pengujian ini, penulis menggunakan lingkaran dengan diameter 4 cm sebagai *template*. Proses *template matching* dilakukan untuk setiap titik dalam lingkaran kaleng yang berjarak kurang dari 4 cm. Dengan membandingkan tiap titik pada semua sampel, kita dapat mengetahui suatu sampel termasuk objek yang OK atau *reject*.

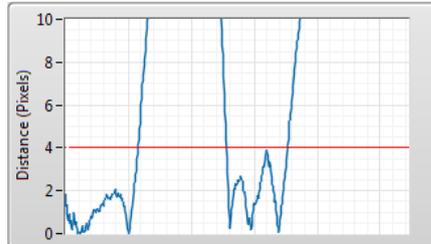
a. Pengujian sample dengan kondisi Good (OK)

Pada pengujian kaleng *Good* (OK), penulis uji sampel sebanyak 3 kali dengan orientasi peletakan objek yang berbeda. Sampel yang diuji ditunjukkan pada gambar 5. Garis merah merupakan jarak diameter objek referensi sedangkan warna biru adalah jarak tiap *pixel* terhadap *template* lingkaran.

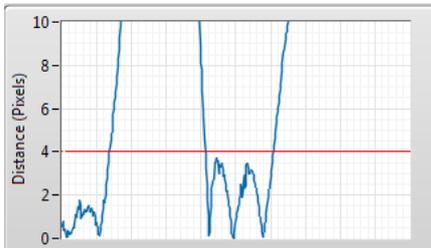
Hasil pengujian ditunjukkan gambar 6. Sumbu *Y* menunjukkan jarak tiap *pixel* dan sumbu *X* menunjukkan posisi titik tersebut pada lingkaran dimulai dari sudut 0–360. Garis merah pada grafik merupakan *threshold* pada tiap titik sebagai pembatas suatu objek mirip dengan objek referensi. Dari ketiga pengujian di atas didapatkan bahwa tidak ada satu nilai dari posisi titik uji pada kaleng yang melebihi *threshold*. Nilai RMS hasil pengujian masing-masing yaitu 0,39; 0,74 dan 0,57 *pixels*. Sehingga rata-rata RMS untuk ketiga pengujian di atas adalah 0,59 *pixels*. Hal ini menunjukkan bahwa sample uji masuk kriteria ok (*GOOD*).

Gambar 7. Sampel uji kaleng *reject1*

a. Pengujian ke 1



b. Pengujian ke 2



c. Pengujian ke 3

Gambar 8. hasil pengujian objek *reject1*

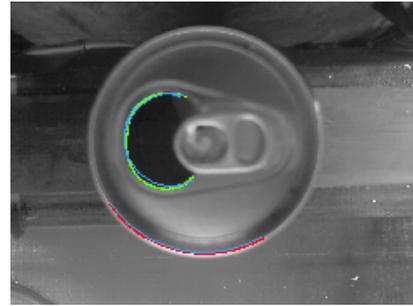
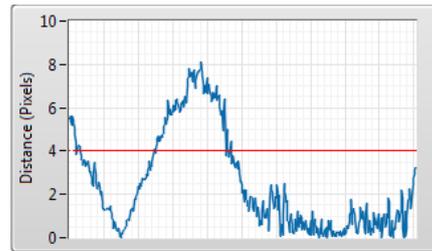
b. Pengujian sample reject 1

Pada pengujian kaleng *reject 1* (kaleng dengan diameter yang tidak sesuai), penulis uji sampel sebanyak 3 kali dengan orientasi peletakan objek yang berbeda. Sampel yang diuji ditunjukkan pada gambar 7. Hasil pengujian ditunjukkan gambar 8.

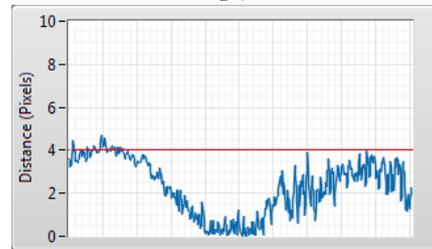
Dari ketiga pengujian di atas didapatkan bahwa ada beberapa titik dalam kaleng yang jarak pixelnya melebihi *threshold*. Nilai RMS hasil pengujian masing-masing yaitu 17,39; 20,87 dan 20,51 *pixels*. Sehingga rata-rata RMS untuk ketiga pengujian di atas adalah 19,59 *pixels*. Hal ini menunjukkan bahwa sampel uji masuk kriteria *reject*.

c. Pengujian sample reject 2

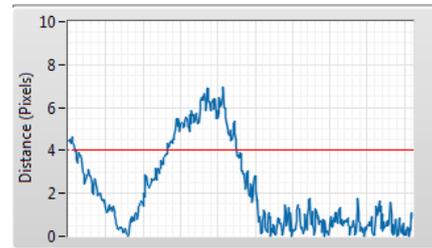
Sampel uji *reject2* (kaleng dengan pembuka kaleng terbuka) dan hasil pengujian ditunjukkan oleh gambar 9 dan gambar 10.

Gambar 9. Sampel uji kaleng *reject2*

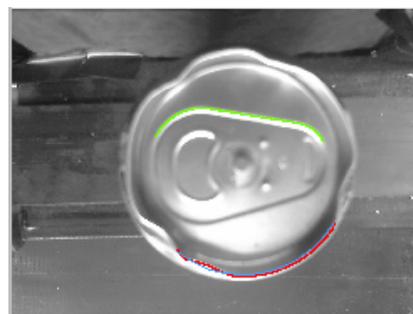
a. Pengujian 1



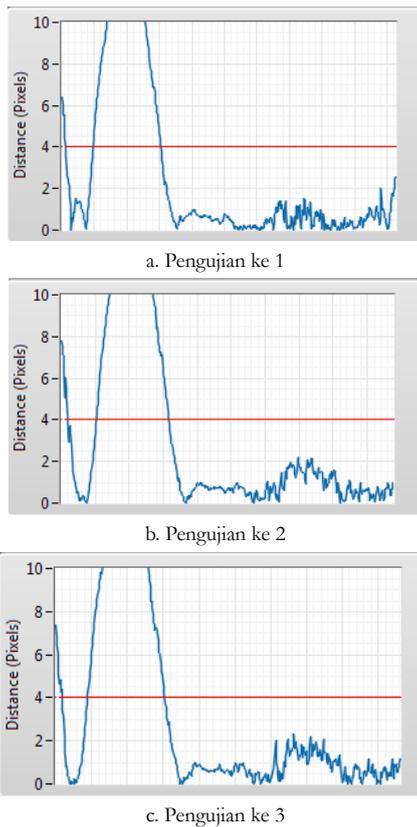
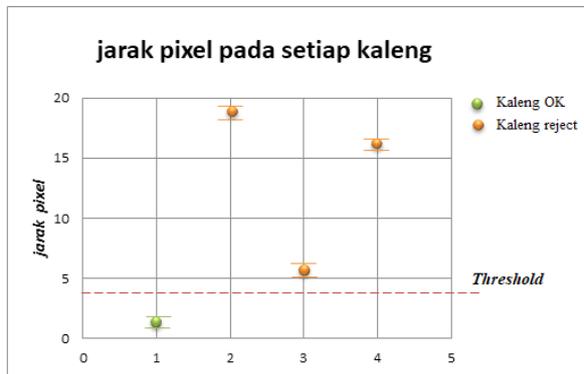
b. Pengujian 2



c. Pengujian 3

Gambar 10. hasil pengujian objek *reject2*Gambar 11. Sampel uji kaleng *reject3*

Ketiga pengujian menunjukkan ada beberapa titik dalam kaleng yang jarak pixelnya melebihi *threshold*. Nilai RMS hasil pengujian masing-masing yaitu 4,72; 5,19 dan 5,24 *pixels*. Rata-rata RMS untuk ketiga pengujian di atas adalah 5,05 *pixels*. Hal ini menunjukkan bahwa sampel uji masuk kriteria *reject*.

Gambar 12. hasil pengujian objek *reject3*Gambar 13. Grafik jarak rata-rata *pixel* pada setiap kaleng pengujian

d. Pengujian sample *reject 3*

Sampel uji *reject3* (kaleng tanpa pembuka kaleng terbuka dan diameter tidak beraturan) dan hasil pengujian ditunjukkan oleh gambar 11 dan gambar 12.

Ketiga pengujian menunjukkan ada beberapa titik dalam kaleng yang jarak *pixel*-nya melebihi *threshold*. Nilai RMS hasil pengujian masing-masing yaitu 14,72, 15,19 dan 15,24 *pixels*. Rata-rata RMS untuk ketiga pengujian di atas adalah 15,05 *pixels*. Hal ini menunjukkan bahwa sampel uji masuk kriteria *reject*.

Dari keempat pengujian yang dilakukan (1 kaleng OK dan 3 kaleng *reject*), didapatkan jarak *pixel* rata-rata untuk masing-masing kategori digambarkan dalam grafik pada gambar 13. Terlihat bahwa untuk kaleng dengan kategori OK, nilai jarak *pixel* berada di bawah nilai *threshold*, sementara 3 kaleng *reject* mempunyai jarak *pixel* di atas *threshold*.

4. Kesimpulan

Penelitian ini menerapkan filter warna HSL untuk mendeteksi objek kaleng dari citra yang tertangkap kamera dan melakukan penklasifikasian objek dalam kategori OK atau *reject* menggunakan *template matching* berdasarkan *contour analysis*. Filter HSL dapat mendeteksi objek kaleng dengan baik dalam kondisi latar yang berubah. *Template matching* berhasil mengklasifikasi objek melalui tiga kali pengujian dengan orientasi dan posisi objek yang berbeda. Objek berhasil diklasifikasi dengan baik dalam dua kategori OK atau *reject*. Dengan menerapkan nilai *threshold* sebesar 4, didapatkan nilai RMS untuk objek kategori OK adalah 0,59. Sementara untuk objek kategori *reject* didapatkan nilai RMS sebesar 19,59; 5,05; dan 15,05 berturut-turut untuk objek *reject* karena diameter tidak beraturan, objek *reject* dengan pembuka kaleng terbuka dan objek *reject* dengan tanpa pembuka kaleng.

Penelitian ini masih perlu ditingkatkan terutama dalam pendeteksian objek yang *reject* sedikit seperti diameter hanya di satu sisi yang berbeda. Hal ini masih memungkinkan dikategorikan objek *good*. Perbaikan perlu dilakukan dengan menambahkan kriteria klasifikasi lainnya seperti *pixel matching*. Ini masih tersisa untuk penelitian selanjutnya.

5. Daftar Pustaka

- [1] V. Mosorov and J. Nowakowski, "Image Defect Detection Methods for Visual Inspection Systems," in Proceeding of International Conference on the Experience of Designing and Application of CAD Systems in Microelectronics, 2007.
- [2] A. Addien and P. W. Laksono, "Analisis Pengendalian Kualitas Coca-Cola Kaleng Menggunakan Statistical Process Control pada PT CCAI Central Java," in Prosiding Seminar dan Konferensi Nasional, pp. 8–9, 2017.
- [3] B. Sugandi, "Deteksi dan Pelacakan Wajah Berdasarkan Warna Kulit Menggunakan Partikel Filter," *J. Rekayasa Elektr.*, vol. 14, no. 2, pp. 116–122, 2017.
- [4] K. Basha, P. Ganesan, V. Kalist, B. S. Sathish, and J. M. Mary, "Comparative Study of Skin Color Detection and Segmentation in HSV and YCbCr Color Space," *Procedia - Procedia Comput. Sci.*, vol. 57, pp. 41–48, 2015.
- [5] R. Wijanarko and N. Eko, "Deteksi Wajah Bebas Segmentasi Warna Kulit Menggunakan Ruang Warna YCbCr dan Template Matching," *J. Ilm. Cendekia Eksakta*, vol. 2, no. 1, pp. 1–6, 2017.
- [6] N. Izzati, N. Anis, M. Razali, and H. Achmad, "Fire Recognition Using RGB and YCbCr Color Space," *J. Eng. Appl. Sci.*, vol. 10, no. 21, pp. 9786–9790, 2015.
- [7] M. Balcilar, M. F. Amasyali, and A. C. Sonmez, "Moving Object Detection using Lab2000HL Color Space with Spatial and Temporal Smoothing," *J. Appl. Math. Inf. Sci.*, vol. 1766, no. 4, pp. 1755–1766, 2014.

- [8] S. Tsai and Y. Tseng, "A novel color detection method based on HSL color space for robotic soccer competition," *Comput. Math. with Appl.*, vol. 64, no. 5, pp. 1291–1300, 2012.
- [9] A. M. Aibinu, A. A. Shafie, and M. J. E. Salami, "Performance Analysis of ANN based YCbCr Skin Detection Algorithm," *Proceeding Int. Symp. Robot. Intell. Sensors*, vol. 41, no. Iris, pp. 1183–1189, 2012.
- [10] N. S. Widodo and A. Rahman, "Vision Based Self Localization for Humanoid Robot Soccer," *Telkonnika*, vol. 10, no. 4, pp. 637–644, 2012.
- [11] E. Budianita and L. Handayani, "Implementasi Pengolahan Citra dan Klasifikasi K- Nearest Neighbour Untuk Membangun Aplikasi Pembeda Daging Sapi dan Babi," *J. Sains, Teknol. dan Ind.*, vol. 12, no. 2, pp. 242–247, 2015.
- [12] G. Sharma, "Image Recognition System using Geometric Matching and Contour Detection," *J. Comput. Appl.*, vol. 51, no. 17, pp. 48–53, 2012.
- [13] B. Leksono, A. Hidayatno, and R. R. Isnanto, "Aplikasi Metode Template Matching untuk Klasifikasi Sidik Jari," *Transmisi*, vol. 13, no. 1, pp. 1–6, 2011.
- [14] M. & S. R. Bianco Simone & Buzzelli, "Object Detection Using Feature-based Template Matching," *Proc. SPIE - Int. Soc. Opt. Eng.*, 2013.
- [15] W. and Y. P. A, "Penggunaan Template Matching untuk Identifikasi Kecacatan pada PCB," in *Seminar Nasional Aplikasi Teknologi Informasi*, 2008.
- [16] I. Pham, R. Jalovecky, and M. Polasek, "Using template matching for object recognition in infrared video sequences," in *Proceeding of IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, p. 8C5-1-8C5-9, 2015.
- [17] A. Banharnsakun and S. Tanathong, "Object Detection Based on Template Matching through Use of Best-So-Far ABC," *Comput. Intel. Neurosci.*, 2014.